

## **Historic, archived document**

Do not assume content reflects current scientific knowledge, policies, or practices.



Q521  
.R44A7  
e2

5



United States  
Department of  
Agriculture

Agricultural  
Research  
Service

ARS-53

September 1986

# A Computer Management System for Apple ("Malus X domestica" Borkh.) Germplasm With Resistance to Disease and Arthropod Pests

SLA  
copy  
13.95  
1986

## ABSTRACT

Goonewardene, H. F., V. Rudkevich, R. Grosso, and E. B. Williams. 1986. A Computer Management System for Apple (Malus X domestica Borkh.) Germplasm With Resistance to Disease and Arthropod Pests. U.S. Department of Agriculture, Agricultural Research Service, ARS-53, 26 p.

This publication describes a computerized system for the storage, processing, and retrieval of available information on 2,384 entries of apple (Malus X domestica Borkh.) progenies, selections, and cultivars used in breeding for resistance to diseases and arthropods and for horticultural characteristics. The procedures that were developed for the CDC computer models 6500 and 6600 using Pascal as the high level language have been modified for use on the Apple IIe personal computer.

KEYWORDS: apple breeding, computer program, Pascal, germplasm, horticultural characteristics, disease resistance, arthropod resistance.

\*\*\*\*\*

Copies of this publication may be purchased from the National Technical Information service, 5285 Port Royal Road, Springfield, VA 22161.

ARS has no additional copies for free distribution.

A COMPUTER MANAGEMENT SYSTEM FOR APPLE ("MALUS X  
DOMESTICA" BORKH.) GERMPLASM WITH RESISTANCE  
TO DISEASES AND ARTHROPOD PESTS

H.F. Goonewardene, V. Rudkevich, R. Grosso, and E.B. Williams<sup>1</sup>

INTRODUCTION

An important contribution to apple production could be made through the development of germplasm with the genetically inherited traits for resistance to one or several economically important pests. This approach, if successful, would minimize the dependence on the multiple applications of several pesticides for producing cosmetically acceptable apples, the impact on pollution of the environment, and the dependence on fossil fuels. A large body of data will necessarily be acquired during such a task, and rapid access to the data base is essential for its exploitation. The feasibility of using contemporary computer technology in handling the necessary information has been reported by Ghaderi et al. (1984). The only available report on computer usage in tree fruits for other than developing an inventory is that of Bell et al. (1980), who developed a COBOL program for tracing pedigrees from a data base developed for pears, Pyrus sp., and those of Van der Zwet et al. (1979) and Blake and Van der Zwet (1979) who developed a data base for use in the U.S. Department of Agriculture's pear breeding program.

We chose Pascal from among the well defined, high-level languages, for our work on coding and programming. Our programs were developed for use with the main-frame computers (CDC6500 and 6600) at Purdue University, West Lafayette, IN.

**The data base, files and programs** are illustrated in figure 1 and described as follows:

---

<sup>1</sup>Goonewardene is a research entomologist with the U.S. Department of Agriculture, Agricultural Research Service; Rudkevich and Grosso are student programmers, and Williams is Professor (Emeritus), Department of Botany and Plant Pathology, all at Purdue University, West Lafayette, IN 47907.

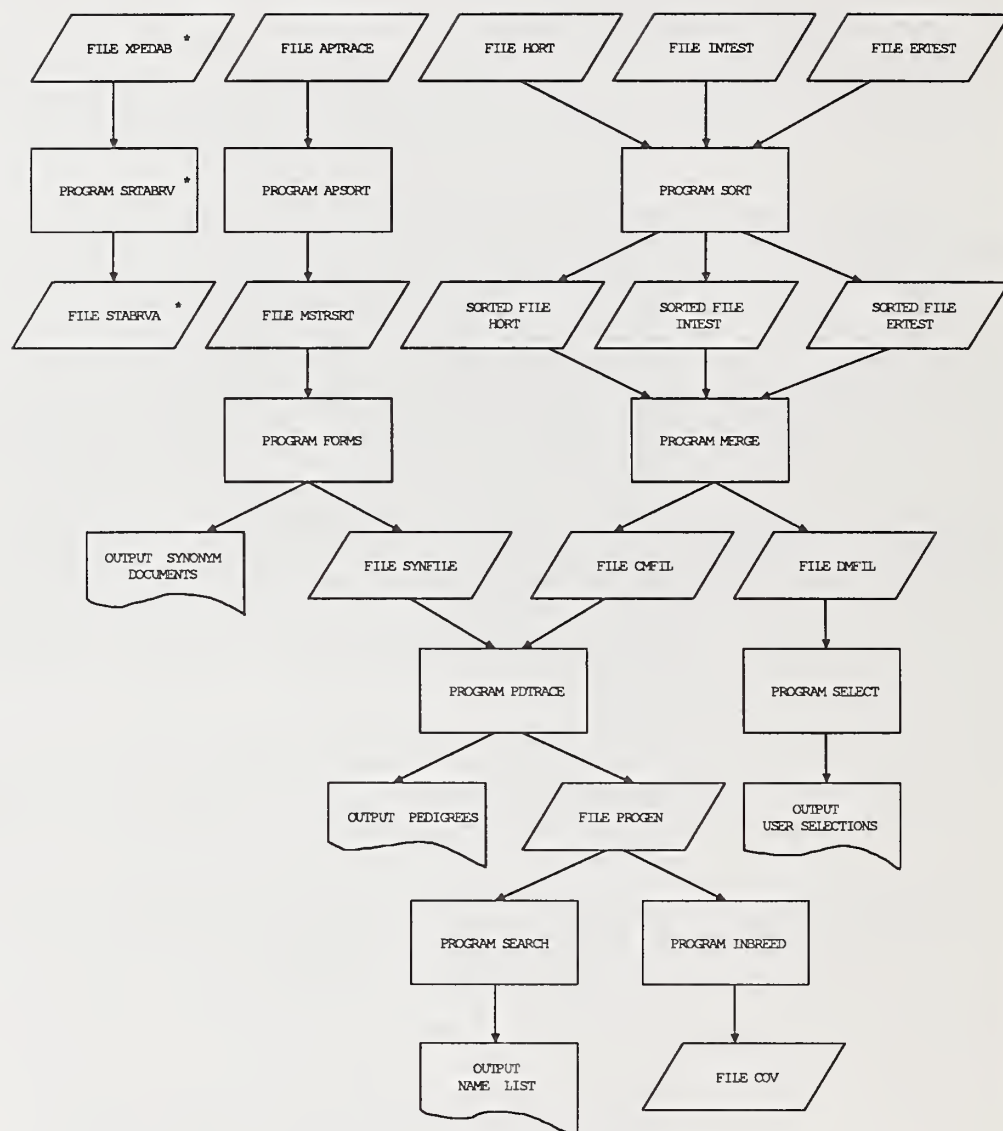


Figure 1.

Flow chart of the general structure of the data base and program sequences for its utility in host resistance.

\* Are user references that store the complete names of all apple selections in this data base.

**The data base** is composed of information recorded over a period of 30+ years and contained in the following files:

XPEDAB  
APTRACE  
HORT  
INTEST  
ERTEST

The contents of each file is described below:

1. File name: **XPEDAB** - an unsorted master file of the complete names of all entries and their corresponding abbreviated names. Each abbreviated name has a limit of 12 characters. Blanks are provided if the abbreviated names have less than the limit of 12 characters. The complete name is limited to 121 characters. Providing blanks to reach this limit is not necessary. An example of the entries in the file **XPEDAB** is as follows:

Column	1	13	16
	Abbreviated name		Complete name
	ANTMON	=	ANTONOVKA MONASIR HENRIETTA CROSBY

Note- An unduplicated, unique name and abbreviation is used for each entry.

2. File name: **APTRACE** - an unsorted master file of progeny/parent combinations that is the source of several other operations. The name of a progeny or a parent is limited to 12 characters. Blanks are provided for names that have less than 12 characters to reach this limit.

An example of the entries in the file **APTRACE** is as follows:

Column	1	13	25
	Progeny	Female parent	Male parent
	HER3T176	GDEL	COOP 17
	PWR18T28	JON	OR53T52

3. File name: **HORT** - an information file on the origin of each germplasm entry, along with the associated horticultural characteristics and its response to disease.



A. Germplasm<sup>2</sup> entries: Each entry is identified in columns 1-10 by (1) name, for example, Jonathan, (2) numeric code, for example 2907, or (3) alphanumeric identification, for example HAR21T200, depending on whether the individual entry was a cultivar, progeny, or selection. Germplasm obtained as a plant introduction is identified by the the prefix PI followed by a letter plus 5 or 6 numbers, for example PIB57270<sup>3</sup> or PI104727. Germplasm entries from the Cooperative apple breeding program among Purdue (P) University, W. Lafayette, IN, Rutgers (R) University, New Brunswick, NJ, and the University of Illinois (I) Champaign-Urbana, IL, (PRI) are identified differently, depending on whether they were progenies or selections. If they were progenies, the identification was by numbers with 2, 3, or 4 digits denoting the progeny or number of the cross, such as 49, or 703 or 1789, followed by a hyphen and a number to identify the seedling, such as 1789-1. Progeny selections were identified after vegetative propagation by the farm and block name composed of one or two letters, 1-3 digit row numbers, and 1-3 digit tree number such as CBR1T29.

Farm and Block listings were as follows:

C = Clark Farm (Blocks A, B, & C; DG = Douglas; H = Hinsley; O = O'Neill Farm (original seedlings); OB = O'Neill Farm (Block B); P = Poultry farm; St = Storage block; T = Throckmorton (Block N = North; Block S = South) and 9A = Block 9A at O'Neill Farm.

The entry CBR1T29 refers to CB = Clark Farm, Block B, R1 = row 1 and T29 = tree 29.

Germplasm originating in New Jersey was identified by the prefix NJ or D1R followed by a maximum of 5 digits or characters, for example, NJ44, D1R8T352, and those from New York had the prefix NY followed by 5 digits with a separating hyphen followed by more digits to indicate the tree numbers, for example NY55166-23.

---

<sup>2</sup>The term "germplasm entry" is used in this paper to indicate a progeny, or a selection or a cultivar.

<sup>3</sup>B57270 and similar entries are B numbers assigned to accessions during the quarantine process before virus indexing, certification, and release. Since these accessions are at the Plant Inventory Stations, we have added the prefix PI to the B numbers for this publication.



B. Horticultural and disease incidence data were coded as shown as shown below:

<u>Column/s</u>	<u>Description</u>
31-32	month of fruiting
33-34	day of fruiting
35-36	year of fruiting
37	size of fruit in inches; coded as 1=2 1/4, 2 = 2 1/2, 3=2 3/4, 4=3.
38	quality; coded as 1=very good, 2=good, 3=fair, 4=poor.
39	shape of fruit; coded as 1=oblate, 2=oblong, 3=conical, 4=round.
40	color of fruit; coded as 1=light red, 2 = medium red, 3=dark red, 4=yellow.
41	flesh type; coded as 1=hard, 2=tough, 3=crisp, 4=tender.
42	season; coded as: 1=late July, 2=early August, 3=late August, 4=early September, 5=late September, 6=October.
43	mildew rating; field rating of mildew coded as 1=highly resistant, 2=resistant, 3=moderately resistant, 4=susceptible, 5=highly susceptible.
44	spur type; coded as 1=yes, 2=no.
45-46	scab resistance; coded for 27 different sources used for introducing resistance to scab (single or multiple genes). 01 = Antonovka, 02 = Antonovka Monassir, 03 = atrosanguinea pit, 04 = atrosanguinea 3 type, 05 = baccata jackii, 06 = floribunda, 07 = English, 08 = Hb1, 09 = Hb2, 10 = Jonsib, 11 = MA4, 12 = MA8, 13 = MA16, 14 = MA1255, 15 = micromalus (pit), 16 = micromalus 3 type, 17 = prunifolia, 18 = prunifolia microcarpa, 19 = prunifolia xanthocarpa, 20 = Russian (differential), 21 = Russian (non-differential), 22 = Sargenti, 23 = sieboldii, 24 = zumi calocarpa, 25 = dwarf spreading toringo, 26 = other; 27 = scab susceptible.

A sample of some entries of the file **HORT** follows:

Column 1	11	21	31	45
<u>Germplasm entry</u>	<u>♀</u>	<u>♂</u>	<u>Coded information</u>	<u>Scab resis- tance</u>
PAR4T236	ROCK24-224	ROCK23-57	101378333226	04
PAR4T8	ROCK25-16	NJ159155	080378334231	06
PAR5T134	RALLS	HAR5T8	1020 321136	06
PAR5T170	RALLS	HAR5T8	100578334236	06
PAR5T176	RALLS	TNR10T11	0928 3231 5	06
PAR5T202	DELICIOUSM	PWR4T5	091878333224	15
PAR5T203	DELICIOUSM	PWR4T5	091878421134	15
PAR8T186	SUMMERRED	PWR14T40	081977324133	06

Note - The priorities in assignment of information by the Pascal Compiler varies depending on the content of the space. In some, a blank space is assigned a lower priority than a space containing a number or a character. Because of this peculiarity in compiling, we see in the previous example that PAR4T8 is listed after PAR4T236.

4. File Name: **INTEST**- information file on the response of the germplasm screened for resistance to codling moth, *Laspyresia pomonella* L. (CM), plum curculio, *Conotrachelus nenuphar* (Herbst) (PC), apple maggot, *Rhagoletis pomonella* (Walsh) (AM), redbanded leafroller, *Argyrotaenia velutinana* (Walker) (RB), according to laboratory procedures described by Goonewardene et al. (1975,1979) is stored in this file. The amount of damage to the fruit for each entry caused by each of these insects and the control (cultivar 'Jonathan') is recorded. The significance of the differences in damage between the control and the germplasm entry is noted as the significance rating (S1 = significant at  $P < 0.05$ , S2 = significant at  $P < 0.01$ , and NS = nonsignificant, that is  $P > 0.05$ ).

An example of the data in the file **INTEST** follows:

Column	1	11	15	19	22	26	29	33	36	40	43	47	50	54	57	62
	Germplasm entry	Year & No. of test	CM stings	Sig.	CM exits	Sig.	PC scars	Sig.	PC exit holes	Sig.	AM stings	Sig.	AM exit holes	Sig.	RB scars	Sig.
	CBR1T29	791	07.0NS		03.0NS											
	CBR2T43	761									00.8NS		00.1NS			
	CBR4T20	781	00.9NS		00.9NS		00.5NS		12.9NS		02.0NS		00.3S1		01.0NS	
	CBR4T20	791	00.0S1		00.0S1											
	CBR4T44	751	02.6NS		03.1NS		01.6NS		02.9NS		03.8NS		02.6NS		00.3NS	
	CCR03T11	761					02.3NS		16.5NS		01.6NS		00.5NS		01.9NS	

Note- CBR4T20 was tested for CM in 1978 and retested in 1979.

5. File name: **ERTEST** - stores information on the mean numbers of European red mites found for an entry compared with those numbers found on HCR21T200 (control) in resistance studies conducted according to the procedures described by Goonewardene et al. (1976). Differences between means between a germplasm entry and the control, HCR21T200, are reported as S1 = significant at  $P < 0.05$ , S2 = significant at  $P < 0.01$ , and NS = nonsignificant that is  $P > 0.05$ .

The example of the entries in the file **ERTEST** is given below:

Column	1	17	23	28
	Germplasm entry	Year of test	Mite count	Significance rating
	HCR21T200	80B	43.3	NS
	HCR23T125	74-75	04.9	S1
	HCR7T163	74-75	03.7	S1
	HCR9T48	80A	01.1	S1
	NJ123056	79	20.0	NS
	NJ123056	80B	20.0	S1
	NJ154955	79	37.5	NS
	NJ154955	80B	37.5	NS
	NJ159155	79	22.8	NS
	NJ159155	80B	22.8	S1
	NJ25	74-75	07.8	NS
	NJ43	79	33.6	NS
	NJ43	80B	33.6	NS

6. Program: **SRTABRV** - sorts the randomly listed, alphanumeric, abbreviated entries from the input file **XPEDAB**. The program outputs a listing of the abbreviated names of entries and their complete names listed alphanumerically.

7. File Name: **STABRVA** - a file containing the output from program **SRTABRV** is organized as follows:

The output below is the example of entries in file **STABRVA**

Column 1	14	16
Germplasm entry <sup>4</sup>		Names of parents
ANNA	=	ANNA
BMDEL	=	BUD MUTATION OF DELICIOUS
EDEL	=	EARLY DELICIOUS // *SYN. WITH EARLIRED DELICIOUS*//
MACOUN NJ28	=	(MACOUN * NJ28)

8. Program: **APSORT** - sorts the randomly listed alphanumeric progeny/parent combinations from the input file **APTRACE** through a binary insertion sorting routine. The output is listed in the file **MSTRSRT**.

9. File name: **MSTRSRT** -file has the output from the program **APSORT** that sorts progeny/parent combinations and lists them by progeny name.

The file follows the same format as was used for **APTRACE**. The data from this file is used as the input for the program **FORMS**.

10. Program: **FORMS** - groups all progenies from the input file **MSTRSRT**, that have identical male and female parents. The output is generalized in two formats: (1) a user readable reference document and (2) a program readable file (See file description **SYNFILE**).

This program initiates a single trace of a pedigree for all progenies having the same pedigree. Traces for progenies with the same pedigree will not be generated but noted after the respective pedigree trace is completed.

<sup>4</sup>In the examples cited, only the abbreviated names of germplasm entries are used; for example (**MACOUN \*NJ28**).

The program reads the file for the name of germplasm as follows:

```

34.000=
35.000=  PROCEDURE READ12(VAR S : NAMES);  (*this*)
36.000=                                         (*procedure*)
37.000=  VAR                                  (*will get*)
38.000=      A :NAMES;                        (*and read 12*)
39.000=      I : INTEGER;                    (*characters*)
40.000=      CH : CHAR;                      (*from the*)
41.000=                                         (*input synfile*)
42.000=  BEGIN
43.000=      FOR I : = 1 TO 12 DO
44.000=          A[I] : = _ _;                (*A designates the*)
45.000=          I : = 0;                    (*germplasm entry*)
46.000=          WHILE (NOT EOLN (SYNFILE)) AND (I < 12) DO
47.000=              BEGIN
48.000=                  I : = I + 1;
49.000=                  READ(SYNFILE CH);
50.000=                  A[I] : = CH;
51.000=              END;
52.000=          S : = A;
53.000=      END;
54.000=
55.000=

```

Note that through the use of Program FORMS one creates a file of synonyms (**SYNFILE**).

11. File Name: **SYNFILE** - lists all entries having identical pedigrees and listing them starting in column 39 as shown in the following example:

Column 1	2	3	39
Germplasm entry	Female parent	Male parent	Names by which known parents have been designated at different stages of previous breeding programs
HER3T50	PWR22T12	PRIMA	HER3T65
HER4T112	PRIMA	PWR22T12	HER4T56
HER4T177	HAR19T194	HCR23T94	HER4T182
HER5T106	HCR6T149	HCR5T78	
HER7T142	TSR13T130	HAR30T106	

12. Program: **SORT**: Sorts randomly listed alphanumeric entries in each of the files, **HORT**, **INTEST**, and



**ERTEST** and arranges the entries in alphabetical order. We used the bubblesort procedure which sorts  $N$  entries at a speed proportional to the square of  $N$  ( $N^2$ ), but if a more rapid sort is desired or if there is a need to sort a very large number of entries, another sorting procedure should be used such as **QUICKSORT** ( $O(N \log N)$ ) as described by Wirth (1976). Sample output of the program **SORT** were shown earlier in files **HORT**, **INTEST**, and **ERTEST**.

13. Program: **MERGE** - the Pascal code for merging the data from the files **HORT**, **INTEST**, and **ERTEST** for each entry and storing it on 1 line of the array<sup>5</sup> **MERGAR** is shown.

```

100.000=PROCEDURE MERGA(VAR LINE : INTEGER);
101.000=VAR                                     (*APPLE-holds*)
102.000=    I,J,K. : INTEGER;                   (*hort*)
103.000=    EXIT : BOOLEAN;                     (*characteristic*)
104.000=                                     (*INSECT-insect*)
105.000=BEGIN                                   (*resistance and*)
106.000=    EXIT : =FALSE;                      (*MITE-mite*)
                                           (*resistance*)
                                           (*information*)
107.000=    WHILE (NOT EXIT) AND (NAMEA[LINE]>=NAMEI[CT]) AND
                                           (CT<N) DO
108.000=    BEGIN
109.000=        IF (NAMEA[LINE]=NAMEI[CT]) THEN (*NAMEA-Entry*)
110.000=        BEGIN                          (*from HORT,*)
111.000=            EXIT : = TRUE;              (*NAMEI-entry*)
112.000=        END;                          (*from*)
113.000=                                           (*INTEST,*)
114.000=        CT : = CT+1;                    (*NAMEI-*)
115.000=    END;                              (*entry*)
116.000=    FOR J : = 1 to 48 DO                (*from*)
                                           (*ERTEST*)
117.000=        MERGAR[LINE][J] : = APPLE[J,LINE]; (*MERGAR-*)
118.000=    IF EXIT THEN                       (*merged*)
119.000=    BEGIN                              (*informa-*)
120.000=        FOR J : = 49 TO 124 DO          (*tion*)
121.000=            MERGAR[LINE][J] : = INSECT[J-38,CT];(*If*)
                                           (*found a*)
122.000=    END                               (*corresponding*)
123.000=    ELSE                               (*entry in*)
124.000=    BEGIN                             (*insect*)
                                           (*information*)

```

---

<sup>5</sup>Compiler restrictions necessitated declaring an array of 350 entries with 200 characters/entry. With our data we were obligated to run the data as 5 operations.

```

(*merge it with*)
(*hort*)
(*information*)
125.000=      NOINS : = TRUE;(*NOINS-no insect info found*)
126.000=      END;
127.000=
128.000=      EXIT : = FALSE;
129.000=      WHILE (NOT EXIT) AND(NAMEA[LINE]>=NAMEM[CT2])
              (CT2<N) DO
130.000=      BEGIN
131.000=          IF (NAMEA[LINE]=NAMEM[CT2]) THEN (*If found*)
132.00=          EXIT : = TRUE;                    (*at corresponding*)
133.00=          CT2 : = CT2+1;                    (*entry in mite*)
134.000=      END;                                (*information*)
135.000=          (*merge*)
136.000=      IF EXIT THEN
137.000=      BEGIN
138.000=          FOR J : = 125 to 159 DO
139.000=              MERGAR[LINE][J] : = MITE[J-108,CT2];
140.000=      END;
141.000=END;
142.000=

```

This program **MERGE** will generate 2 types of output files:

(1) File: **CMFIL** for use as a database for program **PDTRACE**, along with **SYNFILE**.

(2) File: **DMFIL** for use as a reference document and the database for program **SELECT**.

14. File name: **CMFIL**: A master file created by using the program **SORT** to sort and list entries from the file **HORT**, and then merging the data from the **INTEST** and **ERTEST** files into this sorted listing through the use of the program **MERGE**. **CMFIL** is used as the data base in future programs.

The file is organized as follows:

<u>Column</u>	<u>Description</u>
1-10	germplasm entry
11-20	female parent
21-30	male parent
31-46	horticultural information as described for file <b>HORT</b>
49-101	insect information as described for file <b>INTEST</b>
127-146	mite information as described for file <b>ERTEST</b>



An example of the data in the file **CMFIL** follows:

Column	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CBR1T22	MCINTOSH	TNR6T13	082569121243	06	781	01.5NS	01.1NS	00.6NS	01.7NS	01.0NS	00.2NS	07.0NS			
CBR1T29	MCINTOSH	TNR6T13	090269221244	06	781	00.1NS	00.7NS	01.1NS	04.9NS						
CBR2T43	MCINTOSH	TNR6T13	082569123133	06	781	00.9NS	00.9NS	00.5NS	12.9NS	02.0NS	00.3S1	01.0NS			
CBR3T2	597NJ1	NJ355959	082569334233	06											

15. File name: **DMFIL** - lists merged data from files **HORT**, **INTEST** and **ERTEST** in the uncoded and expanded form. The following is an example of information contained in the file.

```

SELECTION : CBR1T22
FEMALE PARENT : MCINTOSH
MALE PARENT: TNR6T13
THE DATE OF TEST : 08/25/69
SIZE: 2.25 INCHES
QUALITY: GOOD
SHAPE: OBLATE
COLOR: MED RED
FLESH TYPE: TENDER
SEASON: LATE AUGUST
MILDEW RATING:
SPUR TYPE: NO
RESISTANCE SOURCES : FLORIBUNDA
THE YEAR OF INSECT TEST: 1977
TEST NUMBER: 1
CODLING MOTH STINGS: 01.5NS
CODLING MOTH EXIT HOLES : 01.1NS
PLUM CURCULIO FEEDING SCARS : 00.6NS
PLUM CURCULIO EXIT HOLES : 01.7NS
APPLE MAGGOT STINGS: 01.0NS
APPLE MAGGOT EXIT HOLES : 00.2NS
RED BANDED LEAFROLLER SCARS : 07.0NS
RED BANDED LARVAE :
RED BANDED ADULTS :
MITE INFORMATION:
+++++
Note:  BLANKS DENOTE NO INFORMATION AVAILABLE

```

(For some entries test data from screening for resistance to insects and/or mites were unavailable; for example, CBR 3T2. To handle such situations, blank spaces were provided in the program.)

When a data set is unavailable, for example insect and/or mite information, the program provides blank spaces, for CBR1T22 information on MILDEW, test results on REDBANDED LARVAE, and REDBANDED ADULTS are lacking. Therefore, spaces reserved for such information are left blank.

16. Program: **SELECT** - Uses file **DMFIL** as the database for input. The program can be used to select individuals and produce an output with the certain characteristics.

- A. (I) with a specified parent
- (II) with specific disease resistance
- (III) with a specified shape
  - or size or color or quality or fleshtype
  - or season of harvest or mildew rating
  - or all of the above
- (IV) with insect and/or mite resistance and a designated level of significance (P < .05 or .01)

An example of an output generated by SELECT is shown below:

```

INDIVIDUALS WITH SIZE> 2.25 INCHES
HAR1T11                      3.00 inches
HAR1T25                      3.00 inches
HAR1T71                      3.00 inches
HAR2T46                      2.75 inches
HAR2T128                    2.5 inches
HAR2T164                    2.5 inches
HAR2T169                    2.75 inches
HAR2T146                    2.5 inches
INDIVIDUALS WITH GOOD QUALITY
HAR1T11
HAR1T25
HAR2T128
HAR2T159
HAR2T178
THE SEASON OF HARVEST IS OCTOBER
HAR6T51
CBR4T10
CBR9T28
  
```

The following is an example of the Pascal code used to select the germplasm entry for a specified property or trait. The procedure, to choose individuals with a desired property, uses a "string" type data. The string describes the property, such as good, poor, and so forth.

```

PROCEDURE CHQUAL(STRING : STYPE);
VAR
  I,J: INTEGER;
BEGIN
  WRITELN(_INDIVIDUALS WITH_,STRING,_QUALITY_);
  FOR I: = 1 TO N DO
    BEGIN
      IF RECARR[I]. QUAL=STRING THEN      (*Recarr-holds*)
        WRITELN(__RECARR[I].APPLE);      (*all of the*)
      END;                                (*character-*)
                                           (*istics*)
                                           (*of the*)
                                           (*germplasm*)
                                           (*example*)
                                           (*RECARR[I].*)
                                           (*Qual=quality*)
    WRITELN;
  END; (*PROCEDURE*)

```

B. (I) From a list of disease resistance germplasm entries this program will also generate:

1. A list of all parents used in these germplasm entries;
2. The parents most frequently occurring parents in the group of entries;
3. Calculate the percentage of times they (parents) occur.

(II) Repeats steps (I) 1., 2., and 3. for entries with pest resistance, for example insect or mite resistance.

The Pascal code for listing the parents in a group of germplasm entries and calculating the percentage frequency of their occurrence is described below:

```

PROCEDURE PERCENT(NAME:STYPE);
VAR
  I,J,K : INTEGER;
  PER : REAL;
BEGIN
  J:=0;
  FOR I:= 1 DO 200 00
    IF COUNTLIST[I]<>0 then (*countlist-holds count of*)
                                           (*how frequent each*)
      J:=J+COUNTLIST[I];  (*parent occurred*)
    I:=1;
    WHILE (I<200) and (PARLIST[I]<>NAME) DO (*PARLIST-list*)
                                           (*of parents*)
    BEGIN
      I:=I+1;
      IF PARLIST[I]=NAME THEN      (*countlist-how*)
                                           (*many times*)
                                           (*each parent*)

```

```

(*occurred*)
K:=COUNTLIST[I];
END;
IF J<>0 THEN
BEGIN
(*per-variable used*)
(*to calculate the*)
PER :=(K/J)*100;
(*percentage*)
WRITELN(_THE PARENT_,NAME,: OCCURS_,PER:5:3,_PERCENT OF
THE LINE_);
END
ELSE
WRITELN(_THE DATA ON PARENTS NAMES IS NOT AVAILABLE_);
END;

```

The sample of the output:

INDIVIDUALS WITH ZUMI CALOCARPA RESISTANCE

-----  
CCR3T18  
CCR3T27  
CCR3T35  
CCR3T44  
CCR3T47  
CCR3T54  
CDR3T3  
CDR5T17  
CDR5T8  
CDR5T9

FEMALE PARENT : MCINTOSH

-----		
CCR3T18	MCINTOSH	TSR4T237
CCR3T27	MCINTOSH	TSR4T237
CCR3T35	MCINTOSH	TSR4T237
CCR3T44	MCINTOSH	TSR4T237
CCR3T47	MCINTOSH	TSR4T237
CCR3T54	MCINTOSH	TSR4T237

MALE PARENT : MCINTOSH

None

-----  
FEMALE PARENT : TNR2T52

None

-----  
MALE PARENT : TNR2T52

-----		
CDR3T3	TSR5T27	TNR2T52
FEMALE PARENT : TSR5T27		
-----		





```

152.000=      FIND(HSDPTR^.PEDSEED,MOTHER,FATHER);
              (*BINARY SEARCH*)
153.000=
154.000=      NEW(NODE);(*initiates pointer to the new*)
              (*germplasm entry*)
155.000=      NODE^. FEMALEPTR : = NIL;(*pointer to the*)
              (*female parent*)
156.000=      NODE^. MALEPTR : = NIL; (*pointer to the*)
              (*male parent*)
157.000=      NODE^. PEDSEED : = MOTHER; (*append new*)
              (*entry to the end*)
              (*of the list*)
159.000=
160.000=      (*SET MALE*)
161.000=      NEW(ONE);          (*do the same for the*)
              (*male parent*)
162.000=      ONE^.FEMALEPTR : = NIL;
163.000=      ONE^.MALEPTR : = NIL;
164.000=      ONE^.PEDSEED : = FATHER;
165.000=      HSDPTR^.MALEPTR : = ONE;
166.000=      IF (MOTHER < > BLANKS) THEN
167.000=      BEGIN
168.000=          PEDIGREE(NODE);
169.000=      END;
170.000=      IF FATHER < > BLANKS THEN
171.000=          PEDIGREE (ONE);
172.000=      END;
173.000=      END;

```

A companion procedure **TRAVERSE** is used to find the names of the ancestors of each germplasm entry and print them in the appropriate position of a "family tree" as dictated by the procedure **PAGEOUT**.

```

254.000=PROCEDURE TRAVERSE (TREE : SDPTR; LEVEL : INTEGER);
255.000=
256.000=
257.000=
258.000=
259.000=
260.000= BEGIN
261.000=     IF TREE < > NIL
262.000=
263.000=     BEGIN
264.000=         TRAVERSE(TREE^.FEMALEPTR, LEVEL+1);
265.000=         IF (TREE^. PEDSEED < > _UNKNOWN _)
266.000=             THEN
267.000=                 IF (TREE^. PEDSEED < > _CHANCE SDLNG_)
268.000=                     THEN
269.000=                         IF (TREE^. PEDSEED< > _
270.000=                             )AND(TREE^.PEDSEED
271.000=                             < > _ANTMOM_) THEN

```

```

268.000=                WRITELN (PROGEN,TREE^.PEDSEED,
                        TREE^.FEMALE PTR^.PEDSEED,
                        TREE^.MALE PTR^.PEDSEED)
269.000=                PAGEOUT (TREE^.PEDSEED,LEVEL);
270.000=                TRAVERSE (TREE^.MALEPTR, LEVEL+1);
271.000=                END
272.000=END;

```

Figure 2 shows a typical pedigree trace generated by this program.

18. File: **PROGEN**: The program **PDTRACE** will also generate the file **PROGEN** in which each germplasm entry with N ancestors (N being a constant) is allocated the required number of lines. Pedigree traces are read from right to left, that is the earliest generation to the most recent. The name of a single ancestor and its known parents are recorded on one line; for example, four ancestors of the cultivar 'ALEXANDER' (including itself) will be represented in the following manner:

```

MBAC
MROBUSTA      MBAC      MPRUN
MPRUN
ALEX          MROBUSTA

```

File **PROGEN** serves as the data-base for the programs **INBREED** and **SEARCH**.

19. Program: **INBREED** - Calculates the covariance values that are statistical measures of interrelationships between individuals in a pedigree (the covariance value may be used to calculate the inbreeding coefficient (F) by the following formula:  $F = \text{COV}(I,J)-1$ . The value of covariance can range from 0 (not related) to 2 (selfed or fully inbred). The covariance is equal to twice the coefficient of coancestry (Falconer 1981) for those who may use the probability approach. The general procedure used for calculating the covariance between two individuals, is shown below:

```

197.000=FOR I : = 1 TO N DO
198.000=  FOR J : = 1 TO N DO
199.000=  BEGIN
200.000=      IF COVAR[I,J] <> 0 THEN
201.000=      BEGIN
202.000=          K : = ICODE [1,J];
203.000=          L : = ICODE [2,J]
204.000=          IF (K<>0) THEN
205.000=          BEGIN
206.000=              COVAR[I,K] : = COVAR[I,J]/2;

```



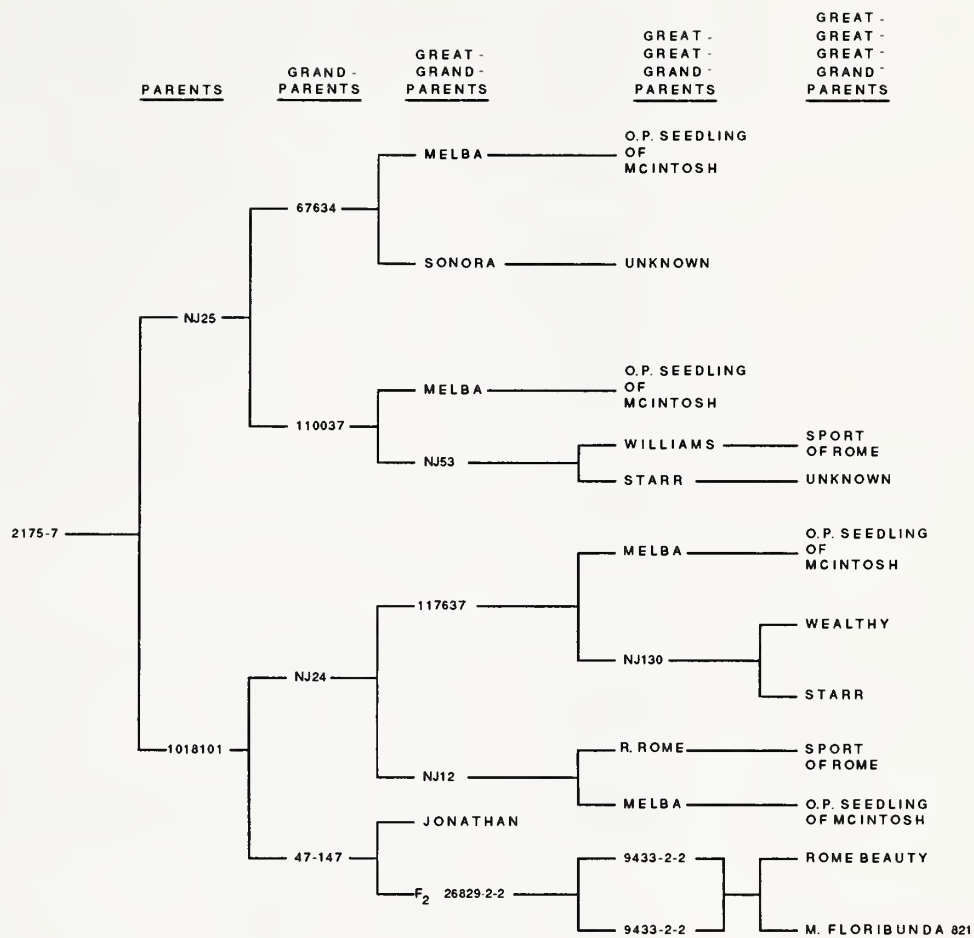


Figure 2.  
Diagram of the pedigree of selection 2175-7  
= HCR21T200=Coop 13 'Redfree' traced back  
5 generations. The seed parent is listed  
first in each cross.

```

207.000=          INBRED : =FALSE;
208.000=          ANSW : =SEARCH (I,INBRED);
209.000=          IF INBRED THEN (*see if individual*)
                        (*is inbred*)
210.000=          COVAR[I,ANSW] : = COVAR[I,ANSW]+1;
211.000=          END ; (*If individual is inbred,use*)
                        (*formula above*)
212.000=          IF (L<>0) THEN
213.000=          BEGIN
214.000=          COVAR[I,L] : = COVAR[I,J]/2;
215.000=          INBRED : = FALSE;
216.000=          ANSW : =SEARCH (I,INBRED);
217.000=          IF INBRED THEN
218.000=          COVAR[I,ANSW] : =
                        COVAR[I,ANSW]+1;
219.000=          END;
220.000=          END;
221.000=          END;

```

I, J, and K are the numeric representations of the alpha-numeric or numeric names of individuals. I and K represent the individuals for which the covariance is to be calculated. COVAR [I,J] is the covariance value between individuals I and J when individual K is related to the individual J. When we consider an inbred individual, the formula changes into  $COVAR [I,I]=F +1$  where  $F = [1/2(COV(I,Female\ Parent) + cov(I,Male\ Parent))] \div \sqrt{(1+F_{Female\ parent_1})(1+F_{Male\ parent_1})}$ .

The program **INBREED** outputs a two-way listing of individuals in the pedigree along with the respective values above 0. The covariance (I, J) = 2F. The alpha-numeric names of individuals are listed both on the first line (row) and in the first column of the table. The covariance values are listed within the table. If there are N individuals in a pedigree, the program will generate X tables ( $X=N/Z$  where N is a variable number of individuals, and Z is a variable number of columns per table, and X is rounded to the next integer. The dimensions of the table thus generated are dependent on the size of the computer screen and/or the size of the paper available for printing out. In our program we used Z=5 which produced an output as shown following:

Germplasm					
Entry	PR14T102	PR7T19	PR7T41	PWR11T146	RMELBA

---

OR48T70	0.50000		
OR48T84	0.50000		
OR52T138			0.50000
OR53T63		0.50000	
PWR14T102	1.00000		
PR7T19		1.00000	
PR7T41			1.00000

Program **INBREED** also generates the list of all individuals and their parents included in the listing as shown below:

Column	1	13	26
	Germplasm entry	FEMALE	MALE
	OR48T70	PR14T104	DEL
	OR48T84	JONRED	PR14T104
	OR52T138	JON	PR7T41
	OR53T52	STARKGL	14-126
	OR53T63	STARKGL	PR7T19
	PI172616		
	PR14T102	76-7	CRAN
	PR7T19	GDEL	F2 26829-2-2
	PR7T41	GDEL	F2 26829-2-2

The presence of a blank entry in the female and/or male parent columns of the listing indicates an unknown parent.

20. File: **COV** - the tables of covariance coefficients and lists of individuals described above are stored in this file.

21. Program: **SEARCH** - Uses file **PROGEN** as a data base to locate the parents of those entries which have common recurring parents. A pedigree is traced for a maximum of five generations and such pedigrees are then searched linearly for recurring parents to indicate inbreeding. The coding used is as described as follows:

```

FOR J:= 1 TO I DO
BEGIN
  B1 :=FALSE;
  B2 :=FALSE;
  FOR K:=J+1 TO I DO
  BEGIN
    IF (PARENT[J]<>_ _ )AND (PARENT[J]<>_CHANCE
      SDLNG_ )THEN
    BEGIN
      B1 :=ALFACHECK(PARENT[J]); (*see if the*)
      (*individual is*)
      (*a cultivar*)
      B2 :=ALFACHECK(PARENT[K]);
      IF B1 AND B2 THEN
        IF (PARENT[J]=PARENT[K]) THEN (*see if the*)
          (*parent*)
          (*occurred more*)
          FOUND :=TRUE; (*than once*)
        END;
      END;
    END;
  IF FOUND THEN
    WRITELN(_ _ ,NAME):
  END

```

The output generated by this program is a list of progeny numbers and/or selections chosen as shown below.

```

646-2
688-100
9AR2T144
9AR4T47

```

The programs described in this publication can be modified for use with Apple IIe or any other personal computer (PC) that can be programmed in Pascal and by making needed changes for program compatability. The reference guide for Apple IIe Pascal describes in detail the characteristics of the unique syntax features to be used for the specific compiler. If used with a PC, the entire programs cited here may be changed to conform to the level of interaction needed between the program and its user.

## REFERENCES CITED

- Ghaderi, A., A.G. Johanson, M.W. Adams and J.D. Kelly. 1984. A computerized system of storage, processing, and retrieval of information for breeding of dry edible beans. *HortScience* 19:345-348.
- Bell, R.L., J. Janick and R.C. Blake. 1980. Computer tracing of pedigrees. *HortScience* 15:303-304.
- Blake, R.C. and T. Van der Zwet. 1979. The USDA pear breeding program III. Fruit evaluation. *Fruit Varieties Journal*. 33:131-136.
- Falconer, D.S. 1960. Introduction to quantitative genetics. 2d edition. Longman, NY. 340 p.
- Goonewardene, H.F., W.F. Kwolek, R.E. Dolphin and E.B. Williams. 1975. Evaluating resistance of apple fruit pests. *HortScience* 10:393-394.
- Goonewardene, H.F., E.B. Williams and W.F. Kwolek. 1976. Resistance to European red mite, *Panonychus ulmi* (Koch), in apple. *Journal American Society of Horticultural Science*. 101:532-537.
- Goonewardene, H.F., W.F. Kwolek, T.E. Mouzin, and E.B. Williams. 1979. A 'No Choice' study for evaluating resistance of apple fruits to four insect pests. *HortScience* 14:165-166.
- Van der Zwet, T., R.C. Blake, and R.H. Zimmerman. 1979. The USDA pear breeding program. II. Seedling evaluation. *Fruit Varieties Journal*. 33:131-136.
- Wirth, N. 1976. Algorithms + Data Structures = programs. Prentice Hall, Inc., Englewood Cliffs, NJ. 366 p.

## ACKNOWLEDGEMENTS

We are indebted to D. Plourde, J. Plourde, R. Slaven, K. Duffy for their efforts in data coding, listing, and entry; G. Smith, B. Blackburn and J. Martin for the development of the files and for several attempts at writing programs. We would also like to thank T. Elmore for his many contributions, and to Terry Stewart, Department of Animal Science, Purdue University, for his advice on the development of the INBREED program. Advice was freely given by C. Geiger and T. McCain, Department of Entomology, Purdue University, on many occasions and we thank them for their contributions. R. Haan played a key role in checking out the programs described in the document and incorporating needed changes or modifications. We thank R.L. Bell, USDA-ARS, NAA, Appalachian Fruit Research Station, Kearneysville, WV, S. Korban, Department of Horticulture, University of Illinois, Champaign-Urbana, and W.E. Nyquist, Department of Agronomy, Purdue University, West Lafayette, IN, for their contributions in a review of an earlier draft of the manuscript.





U. S. DEPT. OF AGRICULTURE  
NATIONAL AGRICULTURAL LIBRARY  
RECEIVED

NOV 7 1986

EXCHANGE UNIT  
ACQUISITIONS SECTION